

CLAIMS

1. A method of formatting a message for exchange between entities on a network, the method comprising:

generating a message envelope;

generating contents of the message envelope, the contents comprising data structures, in which each data structure is identified according to which entity that is intended to process the structure.

2. A method as recited in claim 1, wherein each data structure is further identified according to whether the entity that is intended to process the structure must understand such structure.

3. A method as recited in claim 1, wherein:  
the message envelope has beginning and ending envelope tags;  
the contents of the message envelope is between the envelope tags.

4. A method as recited in claim 1, wherein the contents include:  
a header data structure;  
a body data structure, the body including message data.

5. A method as recited in claim 4, wherein:  
the header data structure has beginning and ending header tags;  
the body data structure has beginning and ending body tags.

1  
2  
3 6. A method as recited in claim 4, wherein:

4 the header data structure is intended for at least one intermediate entity;

5 the body data structure is intended for an destination entity.  
6

7 7. A method as recited in claim 1 further comprising sending the  
8 message envelope to an entity on a network.  
9

10 8. A method as recited in claim 1, wherein at least one of the data  
11 structures includes a request for an entity to perform a task.  
12

13 9. A method as recited in claim 1, wherein the data structures are  
14 expressed in a markup language.  
15

16 10. A method as recited in claim 1, wherein the data structures are  
17 expressed in XML.  
18

19 11. A method as recited in claim 1 further comprising:

20 formatting the message envelope for sending over a network using HTTP;

21 sending the message envelope to an entity on the network by using HTTP.  
22  
23  
24  
25

1       **12.**     A method as recited in claim 1 further comprising:  
2       binding the message envelope into a HTTP request;  
3       sending the message envelope to an entity on the network by using HTTP.

4  
5       **13.**     A method as recited in claim 1 further comprising:  
6       binding the message envelope into a HTTP response;  
7       sending the message envelope to an entity on the network by using HTTP.

8  
9       **14.**     A method as recited in claim 3, wherein the envelope tags identify  
10      the message envelope.

11  
12       **15.**     A method as recited in claim 4, wherein the header tags identify the  
13      header.

14  
15       **16.**     A method as recited in claim 4, wherein the body tags identify the  
16      body.

1           17.    A method as recited in claim 4, wherein the message envelope has  
2 the following format:

3           <Envelope label>  
4                <Header label>  
5                   header data  
6                </Header label>  
7                <Body label>  
8                   message data  
9                </Body label>

10          </Envelope label>

11          the <Envelope label> being the beginning envelope tag, the </Envelope  
12 label> being the ending envelope tag, and the Envelope label identifying the  
13 message envelope;

14          the <Header label> being the beginning header tag, the </Header label>  
15 being the ending header tag, the Header label identifying the header;

16          the <Body label> being the beginning body tag, the </Body label> being  
17 the ending body tag, and the Body label identifying the body;

18          the header data being expressed in XML;

19          the message data being expressed in XML.

20  
21          18.    A computer-readable storage medium having computer-executable  
22 instructions that, when executed by a computer, performs the method as recited in  
23 claim 1.  
24  
25

1 **19.** A method of delivering a message over a network, the method  
2 comprising:

3 transmitting a message envelope from an origin entity to a destination  
4 entity via one or more intermediate entities on the network;

5 the message envelope having contents comprising data structures, in which  
6 each data structure is identified according to which entity that is intended to  
7 process the structure.

8  
9 **20.** A method as recited in claim 19, wherein each data structure is  
10 further identified according to whether the entity that is intended to process the  
11 structure must understand such structure.

12  
13 **21.** A method as recited in claim 19, wherein:  
14 the message envelope has beginning and ending envelope tags;  
15 the contents of the message envelope is between the envelope tags.

16  
17 **22.** A method as recited in claim 17, wherein the contents include:  
18 a header data structure;  
19 a body data structure, the body including message data.

20  
21 **23.** A method as recited in claim 22, wherein:  
22 the header data structure has beginning and ending header tags;  
23 the body data structure has beginning and ending body tags.  
24  
25

Sub  
A1

006030-00000000

24. A method as recited in claim 22, wherein:  
the header data structure is intended for at least one intermediate entity;  
the body data structure is intended for a destination entity.

25. A method as recited in claim 19, wherein at least one of the data structures includes a request for an entity to perform a task.

26. A method as recited in claim 19, wherein at least one of the data structures includes a request for an intermediate entity to perform a task.

27. A method as recited in claim 19, wherein the data structures are expressed in a markup language.

28. A method as recited in claim 19, wherein the data structures are expressed in XML.

29. A method as recited in claim 19 further comprising:  
formatting the message envelope for sending over a network using HTTP;  
sending the message envelope to an entity on the network by using HTTP.

30. A method as recited in claim 19 further comprising:  
binding the message envelope into a HTTP request;  
sending the message envelope to an entity on the network by using HTTP.

31. A method as recited in claim 19 further comprising:  
binding the message envelope into a HTTP response;  
sending the message envelope to an entity on the network by using HTTP.

**32.** A method as recited in claim 21, wherein the envelope tags identify the message envelope.

33. A method as recited in claim 21, wherein the header tags identify the header.

34. A method as recited in claim 21, wherein the body tags identify the body.

1       **35.**    A method as recited in claim 21, wherein the message envelope has  
2 the following format:

3           <Envelope label>  
4                <Header label>  
5                   header data  
6                </Header label>  
7                <Body label>  
8                   message data  
9                </Body label>  
10           </Envelope label>

11       the <Envelope label> being the beginning envelope tag, the </Envelope  
12 label> being the ending envelope tag, and the Envelope label identifying the  
13 message envelope;

14       the <Header label> being the beginning header tag, the </Header label>  
15 being the ending header tag, the Header label identifying the header;

16       the <Body label> being the beginning body tag, the </Body label> being  
17 the ending body tag, and the Body label identifying the body;

18       the header data being expressed in XML;

19       the message data being expressed in XML.

20  
21       **36.**    A computer-readable storage medium having computer-executable  
22 instructions that, when executed by a computer, performs the method as recited in  
23 claim 19.  
24  
25



1        ~~37.~~ A method of parsing a message received by a receiving entity over a  
2 network from an sending entity, the method comprising:

3            parsing a message envelope;

4            parsing contents of the message envelope, the contents comprising data  
5 structures, in which each data structure is identified according to which entity that  
6 is intended to process the structure.

7  
8        38. A method as recited in claim 37, wherein each data structure is  
9 further identified according to whether the entity that is intended to process the  
10 structure must understand such structure.

11  
12        39. A method as recited in claim 38 further comprising if the entity that  
13 is intended to process the structure does not understand such structure, sending a  
14 response message to the sending entity that indicates that the entity did not  
15 understand such structure.

16  
17        40. A method as recited in claim 37 further comprising sending a  
18 response message to the sending entity on the network.

19  
20        41. A method as recited in claim 37, wherein:  
21 the message envelope has beginning and ending envelope tags;  
22 the contents of the message envelope is between the envelope tags.  
23  
24  
25

Sub  
H1

00509076092560

1       **42.**     A method as recited in claim 37, wherein the contents include:  
2       a header data structure;  
3       a body data structure, the body including message data.

4  
5       **43.**     A method as recited in claim 42, wherein:  
6       the header data structure has beginning and ending header tags;  
7       the body data structure has beginning and ending body tags.

8  
9       **44.**     A method as recited in claim 42, wherein:  
10       the header data structure is intended for at least one intermediate entity;  
11       the body data structure is intended for a destination entity.

12  
13       **45.**     A method as recited in claim 37, wherein at least one of the data  
14       structures includes a request for an entity to perform a task.

15  
16       **46.**     A method as recited in claim 37, wherein the data structures are  
17       expressed in a markup language.

18  
19       **47.**     A method as recited in claim 37, wherein the data structures are  
20       expressed in XML.

1       48.    A computer-readable storage medium having computer-executable  
2 instructions that, when executed by a computer, performs the method as recited in  
3 claim 37.

4  
5       49.    A computer-readable storage medium having computer-executable  
6 instructions that, when executed by a computer, performs a method of formatting a  
7 message for exchange between entities on a network, the method comprising:

8           generating a message envelope;

9           generating contents of the message envelope, the contents comprising data  
10 structures, in which each data structure is identified according to which entity that  
11 is intended to process the structure and whether that entity must understand such  
12 structure.

13  
14       50.    A computer-readable storage medium having computer-executable  
15 instructions that, when executed by a computer, performs a method of delivering a  
16 message between entities on a network, the method comprising:

17           transmitting a message envelope from an origin entity to a destination  
18 entity via one or more intermediate entities on the network;

19           the message envelope having contents comprising data structures, in which  
20 each data structure is identified according to which entity that is intended to  
21 process the structure and whether that entity must understand such structure.

1        ~~51.~~ A computer-readable storage medium having computer-executable  
2 instructions that, when executed by a computer, performs a method of parsing a  
3 message received by a receiving entity over a network from an sending entity, the  
4 method comprising:

5            parsing a message envelope;

6            parsing contents of the message envelope, the contents comprising data  
7 structures, in which each data structure is identified according to which entity that  
8 is intended to process the structure and whether that entity must understand such  
9 structure.

10  
11        ~~52.~~ A message exchange apparatus comprising:

12            a processor;

13            a message formatter executable on the processor to:

14                    generate a message envelope;

15            generate contents of the message envelope, the contents comprising  
16 data structures, in which each data structure is identified according to which  
17 entity, over a network of entities, that is intended to process the structure  
18 and whether that entity must understand such structure.

19  
20        ~~53.~~ A message exchange apparatus comprising:

21            a processor;

22            a message deliverer executable on the processor to:

23                    transmit a message envelope from an origin entity to a destination  
24 entity via one or more intermediate entities on the network;  
25

1       54.     the message envelope having contents comprising data structures, in  
2     which each data structure is identified according to which entity, over a network of  
3     entities, that is intended to process the structure and whether that entity must  
4     understand such structure. A message exchange apparatus comprising:

5         a processor;

6         a message parser executable on the processor to:

7             parse a message envelope;

8             parse contents of the message envelope, the contents comprising  
9     data structures, in which each data structure is identified according to which  
10    entity, over a network of entities, that is intended to process the structure  
11    and whether that entity must understand such structure.

12  
13       55.     A message envelope generated in accordance with the following  
14    acts:

15         providing a sending entity in communication with a network of entities;

16         receiving data intended for an intermediate entity;

17         receiving data intended for a destination entity;

18         generating contents of the message envelope, the contents comprising:

19             a header data structure which identifies the intermediate entity as  
20         that which is intended to process the header data structure and whether that  
21         intermediate entity must understand such structure; and

22             a body data structure which identifies the destination entity as that  
23         which is intended to process the body data structure.  
24  
25

1           56. A message envelope as recited in claim 55, wherein the data  
2 structures are expressed in a markup language.

3  
4           57. A message envelope as recited in claim 55, wherein the data  
5 structures are expressed in XML.

6  
7           58. A modulated data signal having computer-executable instructions  
8 embodied thereon comprising:

9           a header data structure which identifies an intermediate entity, over a  
10 network of entities, as that which is intended to process the header data structure  
11 and whether that intermediate entity must understand such structure; and

12           a body data structure which identifies the destination entity as that which is  
13 intended to process the body data structure.

14  
15           59. A modulated data signal as recited in claim 58, wherein the data  
16 structures are expressed in a markup language.

17  
18           60. A modulated data signal as recited in claim 58, wherein the data  
19 structures are expressed in XML.  
20  
21  
22  
23  
24  
25

1        **61.** A computer-readable medium having a data structure embodied  
2 thereon comprising:

3            a header data-structure section which identifies an intermediate entity, over  
4 a network of entities, as that which is intended to process the header data-structure  
5 section and whether that intermediate entity must understand such data-structure  
6 section; and

7            a body data-structure section which identifies the destination entity as that  
8 which is intended to process the body data-structure section.

9  
10        **62.** A computer-readable medium as recited in claim 61, wherein the  
11 data-structure sections are expressed in a markup language.

12  
13        **63.** A computer-readable medium as recited in claim 61, wherein the  
14 data-structure sections are expressed in XML.

15  
16  
17        Add  
18 AI  
19  
20  
21  
22  
23  
24  
25